

# Complexity and Approximation of the Continuous Network Design Problem

Martin Gairing<sup>1</sup>, Tobias Harks<sup>2</sup>, and Max Klimm<sup>3</sup>

1 Department of Computer Science, University of Liverpool, UK  
m.gairing@liverpool.ac.uk

2 Dept. of Quantitative Economics, Maastricht University  
t.harks@maastrichtuniversity.nl

3 Institut für Mathematik, Technische Universität Berlin, Germany  
klimm@math.tu-berlin.de

---

## Abstract

We revisit a classical problem in transportation, known as the *continuous (bilevel) network design problem*, CNDP for short. Given a graph for which the latency of each edge depends on the ratio of the edge flow and the capacity installed, the goal is to find an optimal investment in edge capacities so as to minimize the sum of the routing cost of the induced Wardrop equilibrium and the investment cost for installing the capacity. While this problem is considered as challenging in the literature, its complexity status was still unknown. We close this gap showing that CNDP is strongly NP-complete and APX-hard, both on directed and undirected networks and even for instances with affine latencies. As for the approximation of the problem, we first provide a detailed analysis for a heuristic studied by Marcotte for the special case of *monomial* latency functions (Math. Program., Vol. 34, 1986). We derive a closed form expression of its approximation guarantee for *arbitrary* sets of latency functions. We then propose a different approximation algorithm and show that it has the same approximation guarantee. However, we show that using the better of the two approximation algorithms results in a strictly improved approximation guarantee for which we derive a closed form expression. For affine latencies, e.g., this algorithm achieves a  $49/41 \approx 1.195$ -approximation which improves on the  $5/4$  that has been shown before by Marcotte. We finally discuss the case of hard budget constraints on the capacity investment.

**1998 ACM Subject Classification** F.2 Analysis of Algorithms and Problem Complexity

**Keywords and phrases** Bilevel optimization, Optimization under equilibrium constraints, Network design, Wardrop equilibrium, Computational complexity, Approximation algorithms

**Digital Object Identifier** 10.4230/LIPIcs.xxx.yyy.p

## 1 Introduction

The continuous network design problem (CNDP) introduced by Dafermos [7], Dantzig et al. [9], and Abdulaal et al. [1] is one of the most classical network design problems in transport. In a nutshell, given a graph in which the latency of each edge depends on the ratio of the edge flow and the capacity installed at that edge, the goal is to find an optimal investment in edge capacities so as to minimize the sum of the routing cost of the induced Wardrop equilibrium and the investment cost for installing the capacity. The investment cost is assumed to be linear in chosen capacity and comprises all monetary costs for building the streets with the given capacity (spread over the expected lifespan of the street) and the accumulated maintenance cost during that time. By a scaling capacity costs accordingly, an arbitrary linear combinations of routing cost and investment cost can be minimized. Continuous



© M. Gairing, T. Harks and M. Klimm;  
licensed under Creative Commons License CC-BY

Conference title on which this volume is based on.

Editors: Billy Editor and Bill Editors; pp. 1–16



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

network design is a fundamental problem in traffic and telecommunication networks when new networks have to be designed from scratch, e.g., after introducing new technology or after opening up new areas.

From a mathematical perspective, CNDP is a bilevel optimization problem (cf. [5, 16] for an overview), where in the upper level the edge capacities are determined and, given these capacities, in the lower level the flow will settle into a Wardrop equilibrium in which, for each commodity, only shortest paths are used. Clearly, the lower level reaction depends on the upper level decision because altering the capacity investment on a subset of edges may result in revised route choices by users.

CNDP has been intensively studied since the late sixties (cf. [7, 17]) and several heuristic approaches have been proposed since then; see Yang et al. [27] for a comprehensive survey. Most of the proposed heuristics are numerical in nature and involve iterative computations of relaxations of the problem (for instance the iterative optimization and assignment algorithm as described in [19] and augmented Lagrangian methods or linearizations of the objective in the leader and follower problem). An exception is the work of Marcotte [18] who considered several algorithms based on solutions of associated convex optimization problems which can be solved in polynomial time [11]. He derives worst-case bounds for his heuristics and, in particular, for affine latency functions he devises an approximation algorithm with an approximation factor of  $5/4$ . For general monomial latency functions plus a constant (including the latency functions used by the Bureau of Public Roads [25]) he obtains a polynomial time 2-approximation. Variants of CNDP were also considered in the networking literature, see [4, 12, 13, 14]. These works, however, consider the case where a budget capacity must be distributed among a set of edges to improve the resulting equilibrium. Most results, however, only work for simplified network topologies (e.g., parallel links) or special latency functions (e.g.,  $M/M/1$  latency functions).

### 1.1 Our Results and Used Techniques.

Despite more than forty years of research, to the best of our knowledge, the computational complexity status of CNDP is still unknown. We close this gap as we show that CNDP is strongly NP-complete and APX-hard, both on directed and undirected networks and even for instances with affine latencies of the form  $S_e(v_e/z_e) = \alpha_e + \beta_e \cdot (v_e/z_e)$ , where  $v_e$  is the flow and  $z_e$  the capacity of edge  $e$  and  $\alpha_e, \beta_e \geq 0$ . For the proof of the NP-hardness, we reduce from 3-SAT. The reduction has the property that in case that the underlying instance of 3-SAT has a solution the cost of an optimal solution is equal to the minimal cost of a relaxation of the problem, in which the equilibrium conditions are relaxed. The key challenge of the hardness proof is to obtain a lower bound on the optimal solution when the underlying 3-SAT instance has no solution. To this end, we relax the equilibrium conditions only partially which enables us to bound the cost of an optimal solution from below by solving an associated constrained quadratic optimization problem. With a more involved construction and a more detailed analysis, we can even prove APX-hardness of the problem. Here, we reduce from a symmetric variant of MAX-3-SAT, in which all literals occur exactly twice. While all our hardness proofs rely on instances with an arbitrary number of commodities and respective sinks, we show that for instances in which all commodities share a common sink, CNDP can be solved to optimality in polynomial time.

In light of the hardness of CNDP, we focus on approximation algorithms. We first consider a polynomial time algorithm proposed by Marcotte [18]. This algorithm, which we call BRINGTOEQUILIBRIUM, first computes a relaxation of CNDP by removing the equilibrium conditions. Then, it reduces the edge capacities individually such that the flow

computed in the relaxation becomes a Wardrop equilibrium. We give a novel closed form expression of the performance of this algorithm with respect to the set  $\mathcal{S}$  of allowed latency functions. Specifically, we show that this algorithm is a  $(1 + \mu(\mathcal{S}))$ -approximation, where  $\mu(\mathcal{S}) = \sup_{S \in \mathcal{S}, x \geq 0, \gamma \in [0,1]} \gamma \cdot (1 - S(\gamma x)/S(x))$ . The value  $\mu(\mathcal{S})$  has been used before by Correa et al. [6] and Roughgarden [21] in the context of price of anarchy bounds for selfish routing where they showed that the routing cost of a Wardrop equilibrium is no more than a factor of  $1/(1 - \mu(\mathcal{S}))$  away of the cost of a system optimum. For the special case that  $\mathcal{S}$  is the set of polynomials with non-negative coefficients and maximal degree  $\Delta$ , we derive exactly the approximation guarantees that Marcotte obtained for monomials. As an outcome of our more general analysis, we further derive that this algorithm is a 2-approximation for general convex latency functions and a 5/4-approximation for concave latency functions.

We then propose a new algorithm which we call SCALEUNIFORMLY. This algorithm first computes an optimal solution of the relaxation (as before) and then *uniformly* scales the capacities with a certain parameter  $\lambda(\mathcal{S})$  that depends on the class of allowable latency functions  $\mathcal{S}$ . Based on well-known techniques using variational inequalities (Correa et al. [6] and Roughgarden [21]), we prove that this algorithm also yields a  $(1 + \mu(\mathcal{S}))$ -approximation. As our main result regarding approximation algorithms, we show that using the better of the two solutions returned by BRINGTOEQUILIBRIUM and SCALEUNIFORMLY yields strictly better approximation guarantees. We give a closed form expression for the new approximation guarantee (as a function of  $\mathcal{S}$ ) that, perhaps interestingly, depends not only on the well-known value  $\mu(\mathcal{S})$  but also on the argument maximum  $\gamma(\mathcal{S})$  in the definition of  $\mu(\mathcal{S})$ . We demonstrate the applicability of this general bound by showing that it achieves a 9/5-approximation for  $\mathcal{S}$  containing arbitrary convex latencies. For affine latencies it achieves a  $49/41 \approx 1.195$ -approximation improving on the 5/4 of Marcotte. An overview of our results compared to those of Marcotte can be found in Table 1.

Some proofs missing in this extended abstract can be found in the full version.

■ **Table 1** Approximation guarantees of the algorithms BRINGTOEQUILIBRIUM, SCALEUNIFORMLY, and the best of the two for convex latency functions, concave latency functions and sets of polynomials with non-negative coefficients depending on the maximal degree  $\Delta$ . The approximation guarantees stated for convex latency functions even hold for sets of semi-convex latency functions as in Assumption 2.1. For BRINGTOEQUILIBRIUM, the approximation guarantees marked with (\*) have been obtained before in [18].

Functions	Approximation guarantees	
	BRINGTOEQUILIBRIUM SCALEUNIFORMLY	Better of the two
concave	5/4 = 1.25	49/41 $\approx$ 1.195
convex	2	9/5 = 1.8
polynomials $\Delta$		
0	1	1
1/4	3381/3125 $\approx$ 1.082	$\approx$ 1.064
1/3	283/256 $\approx$ 1.105	$\approx$ 1.083
1/2	31/27 $\approx$ 1.148	1849/1657 $\approx$ 1.116
1	5/4 = 1.25 *	49/41 $\approx$ 1.195
2	$1 + \frac{2}{9}\sqrt{3} \approx 1.385^*$	$\frac{311}{479} + \frac{180}{479}\sqrt{3} \approx 1.300$
3	$1 + \frac{3}{16}\sqrt[3]{4^2} \approx 1.472^*$	$\approx$ 1.369
4	$1 + \frac{4}{25}\sqrt[4]{5^3} \approx 1.535^*$	$\approx$ 1.418
$\infty$	2 *	9/5 = 1.8

## 1.2 Further Application.

One of the most prominent and popular functions used in actual traffic models are the ones put forward by the Bureau of Public Roads (BPR) [25]. They are of the form  $S_e(v_e) = t_e \cdot (1 + b_e \cdot (v_e/z_e)^4)$ , where  $v_e$  is the edge flow,  $t_e$  represents the free-flow travel time,  $b_e > 0$  is an edge-specific bias, and  $z_e$  represents the street capacity, e.g., in terms of the number of lanes and their width. Our best-of-two approximation algorithm yields an improved approximation factor for functions of this type (cf. Table 1 in the appendix) and can be employed to design road networks with a good tradeoff between construction cost and travel times.

Our results have impact beyond this classical application of designing street capacities. Also in telecommunication networks, Wardrop equilibria appear both in systems with source-routing as end-users choose least-delay paths, and in systems with distributed delay-based routing protocols such as OSPF when using the delay for setting the routing weights [26]. The latency at switches and routers depends on the installed capacity and has been modeled by functions of the form  $S_e(v_e/z_e) = \rho (1 + 0.15 (v_e/z_e))^4$ , where  $\rho$  represents the propagation delay and  $z_e$  the installed capacity [20]. These functions fit into our framework, and our analysis improves the state-of-the-art to a 1.418-approximation and can be applied in scenarios where entire new networks have to be designed from scratch, e.g., after introducing new technology such as optical fiber cables. Our 9/5-approximation also applies to Davidson latency functions of the form  $S_e(\frac{v_e}{z_e}) = \frac{v_e}{z_e} / (1 - \frac{v_e}{z_e}) = v_e / (z_e - v_e)$ , where  $z_e$  represents the capacity of edge  $e$ .

## 1.3 Further Related Work.

Quoting [27], CNDP has been recognized to be “one of the most difficult and challenging problems in transport” and there are numerous works approaching this problem. In light of the substantial literature on heuristics for CNDP, we refer the reader to the survey papers [5, 10, 17, 27].

While to the best of our knowledge prior to this work, the complexity status of CNDP was open, there have been several papers on the complexity of the *discrete (bilevel) network design problem*, DNDP for short, see [15, 22]. Given a network with edge latency functions and traffic demands, a basic variant of DNDP is to decide which edges should be removed from the network to obtain a Wardrop equilibrium in the resulting sub-network with minimum total travel time. This variant is motivated by the classical Braess paradox, where removing an edge from the network may improve the travel time of the new Wardrop equilibrium. Roughgarden [22] showed that DNDP is strongly NP-hard and that there is no  $(\lfloor n/2 \rfloor - \epsilon)$ -approximation algorithm (unless  $P = NP$ ), even for single-commodity instances. He further showed that for single-commodity instances the trivial algorithm of not removing any edge from the graph is essentially best possible and achieves a  $\lfloor n/2 \rfloor$ -approximation. For affine latency functions, the trivial algorithm gives a 4/3-approximation (even for general networks) and this is also shown to be best possible. These results in comparison to ours highlight interesting differences. While DNDP is not approximable by any constant for convex latencies, for CNDP we give a 9/5-approximation. Moreover, all hardness results for DNDP already hold for single-commodity instances, while for CDNP we show that this case is solvable in polynomial time.

In independent work, Bhaskar et al. [4] studied a variant of CNDP where initial edge capacities are given and additional budget must be distributed among the edges to improve the resulting equilibrium. Among other results they show that the problem is NP-complete

in single-commodity networks that consist of parallel links in series. This again stands in contrast to our polynomial-time algorithm for CDNP for these instances.

## 2 Preliminaries

Let  $G = (V, E)$  be a directed or undirected graph,  $V$  its set of vertices and  $E \subseteq V \times V$  its set of edges. We are given a set  $K$  of *commodities*, where each commodity  $k$  is associated with a triple  $(s_k, t_k, d_k) \in V \times V \times \mathbb{R}_{>0}$ , where  $s_k \in V$  is the *source*,  $t_k \in V$  the *sink* and  $d_k$  the *demand* of commodity  $k$ . A multi-commodity flow on  $G$  is a collection of non-negative flow vectors  $(\mathbf{v}^k)_{k \in K}$  such that for each  $k \in K$  the flow vector  $\mathbf{v}^k = (v_e^k)_{e \in E}$  satisfies the flow conservation constraints  $\sum_{u \in V: (s_k, u) \in E} v_{s_k, u}^k = \sum_{u \in V: (u, t_k) \in E} v_{u, t_k}^k = d_k$  and  $\sum_{u \in V: (u, w) \in E} v_{(u, w)}^k - \sum_{u \in V: (w, u) \in E} v_{(w, u)}^k = 0$  for all  $w \in V \setminus \{s_k, t_k\}$ . Whenever we write  $\mathbf{v}$  without a superscript  $k$  for the commodity, we implicitly sum over all commodities, i.e.,  $v_e = \sum_{k \in K} v_e^k$  and  $\mathbf{v} = (v_e)_{e \in E}$ . We call  $v_e$  an *edge flow*. The set of all feasible edge flows will be denoted by  $\mathcal{F}$ .

The latency of each edge  $e$  depends on the installed capacity  $z_e \geq 0$  and the edge flow  $v_e$  on  $e$ , and is given by a latency function  $S_e : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$  that maps  $v_e/z_e$  to a latency value  $S_e(v_e/z_e)$ , where we use the convention that  $S_e(v_e/z_e) = \infty$  whenever  $z_e = 0$ . Throughout this paper, we assume that the set of allowable latency functions is restricted to some set  $\mathcal{S}$  and we impose the following assumptions on  $\mathcal{S}$ .

► **Assumption 2.1.** The set  $\mathcal{S}$  of allowable latency functions only contains continuously differentiable and semi-convex functions  $S$  such that the functions  $x \mapsto S(x)$  and  $x \mapsto x^2 S'(x)$  are strictly increasing and unbounded.

Assumption 2.1 is more general than requiring that all latency functions are strictly increasing and convex. For instance, the function  $S(x) := \sqrt{x}$  satisfies Assumption 2.1 although it is concave.

Given a vector of capacities  $\mathbf{z} = (z_e)_{e \in E}$ , the latency of each edge  $e$  only depends on the edge flow  $v_e$ . Under these conditions, there exists a Wardrop flow  $\mathbf{v} = (v_e)_{e \in E}$ , i.e., a flow in which each commodity only uses paths of minimal latency. It is well known (cf. [3, 8, 24]) that each Wardrop flow is a solution to the optimization problem  $\min_{\mathbf{v} \in \mathcal{F}} \sum_{e \in E} \int_0^{v_e} S_e(t/z_e) dt$ , and satisfies the *variational inequality*

$$\sum_{e \in E} S(v_e/z_e)(v_e - v'_e) \leq 0 \quad (2.1)$$

for every feasible flow  $\mathbf{v}' \in \mathcal{F}$ . For a vector of capacities  $\mathbf{z}$  we denote by  $\mathcal{W}(\mathbf{z})$  the corresponding set of Wardrop flows  $\mathbf{v}(\mathbf{z})$ . Beckmann et al. [3] showed that Wardrop flows and optimum flows are related:

► **Proposition 2.1 (Beckmann et al. [3]).** Let  $S_e^*(x) = (xS_e(x))' = S_e(x) + xS_e'(x)$  be the *marginal cost function* of edge  $e \in E$ . Then  $\mathbf{v}^*$  is an optimum flow with respect to the latency functions  $(S_e)_{e \in E}$  if and only if it is Wardrop flow with respect to  $(S_e^*)_{e \in E}$ .

In the continuous (bilevel) network design problem (CNDP) the goal is to buy capacities  $z_e$  at a price per unit  $\ell_e > 0$  so as to minimize the sum of the construction cost  $C^Z(\mathbf{v}, \mathbf{z}) = \sum_{e \in E} z_e \ell_e$  and the routing cost  $C^R(\mathbf{v}, \mathbf{z}) = \sum_{e \in E} S_e(v_e/z_e) v_e$  of a resulting Wardrop equilibrium  $\mathbf{v}$ . Observe that  $C^R(\mathbf{v}, \mathbf{z})$  is well defined as, by (2.1), it is the same for all Wardrop equilibria with respect to  $\mathbf{z}$ . Denote the combined cost by  $C(\mathbf{v}, \mathbf{z}) = C^R(\mathbf{v}, \mathbf{z}) + C^Z(\mathbf{v}, \mathbf{z})$ . We would like to reiterate that other linear combinations can be handled by scaling the capacity prizes accordingly.

► **Definition 1** (Continuous network design problem (CNDP)). Given a directed graph  $G = (V, E)$  and for each edge  $e$  a latency function  $S_e$  and a construction cost  $\ell_e > 0$ , the continuous network design problem (CNDP) is to determine a non-negative capacity vector  $\mathbf{z} = (z_e)_{e \in E}$  that minimizes

$$\min_{\mathbf{z} \geq 0} \min_{\mathbf{v} \in \mathcal{W}(\mathbf{z})} \sum_{e \in E} (S_e(v_e/z_e) v_e + z_e \ell_e). \quad (\text{CNDP})$$

Relaxing the condition that  $\mathbf{v}$  is a Wardrop equilibrium in (CNDP), we obtain the following relaxation of the continuous network design problem:

$$\min_{\mathbf{z} \geq 0} \min_{\mathbf{v} \in \mathcal{F}} \sum_{e \in E} (S_e(v_e/z_e) v_e + z_e \ell_e). \quad (\text{CNDP}')$$

Marcotte [18] showed that for convex and unbounded latency functions, the relaxed problem (CNDP') can be solved efficiently by performing  $|K|$  independent shortest path computations on the graph  $G$ , one for each commodity  $k \in K$ . The following proposition slightly generalizes his result to arbitrary, not necessarily convex latency functions that satisfy Assumption 2.1.

► **Proposition 2.2** (Marcotte [18]). The relaxation (CNDP') can be solved by performing  $|K|$  shortest path computations in polynomial time.

► **Remark.** To speak about polynomial algorithms and hardness, we need to specify how the instances of CNDP, in particular the latency functions, are encoded, cf. [2, 11, 22]. While our hardness results hold even if all functions are linear and given by their rational coefficients, for our approximation algorithms, we require that we can solve (symbolically) equations involving a latency function and its derivative, e.g., Equation (4.4). Without this assumption, we still obtain the claimed approximation guarantees within arbitrary precision by polynomial time algorithms.

### 3 Hardness

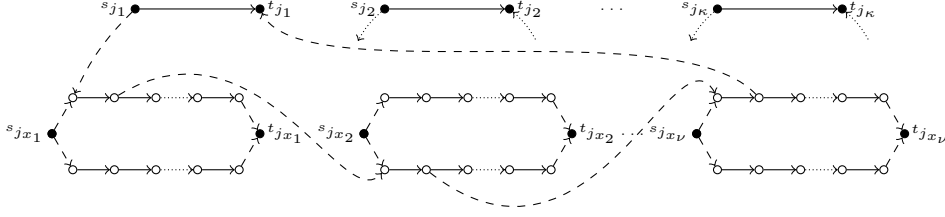
As the main result of this section, we show that CNDP is APX-hard both on directed and undirected networks and even for affine latency functions. The proof of this result is technically quite involved, and we first show the weaker result that CNDP on directed networks is NP-complete.

► **Theorem 2.** *The continuous network design problem (CNDP) on directed networks is NP-complete in the strong sense, even if all latency functions are affine.*

**Proof.** CNDP lies in NP as a vector of capacities  $\mathbf{z}$  is a polynomial certificate. Given  $\mathbf{z}$ , we can compute in polynomial time a corresponding Wardrop equilibrium and the total cost  $C(\mathbf{v}, \mathbf{z})$ .

To show the NP-hardness of the problem, we reduce from 3-SAT. Let  $\phi$  be a Boolean formula in conjunctive normal form. We denote the set of variables and clauses of  $\phi$  with  $V(\phi)$  and  $K(\phi)$ , respectively, and set  $\nu = |V(\phi)|$  and  $\kappa = |K(\phi)|$ . The set  $L(\phi)$  of literals of  $\phi$  contains for each variable  $x_i \in V(\phi)$  the positive literal  $x_i$  and the negative literal  $\bar{x}_i$ , i.e.,  $L(\phi) = \{x_i \in V(\phi)\} \cup \{\bar{x}_i : x_i \in V(\phi)\}$ . In the following, we will associate clauses with the set of literals that they contain.

We now explain the construction of a continuous network design problem based on  $\phi$  that has the property that, for some  $\epsilon \in (0, 1/8)$ , an optimal solution has total cost less or equal to  $(4 + \epsilon)\kappa + 2\kappa\nu$  if and only if  $\phi$  has a solution. Let  $\epsilon \in (0, 1/8)$  be arbitrary. For each clause



■ **Figure 1** Network used to show the hardness of the continuous network design problem. Clause 1 is equal to  $x_1 \vee \bar{x}_2 \vee x_\nu$ . Dashed edges have zero latency.

$k \in K(\phi)$ , we introduce a *clause edge*  $e_k$  with latency function  $S_{e_k}(v_{e_k}/z_{e_k}) = 4 + v_{e_k}/z_{e_k}$  and construction cost  $\ell_{e_k} = (\epsilon/2)^2$ . For each literal  $l \in L(\phi)$  and each clause  $k \in K(\phi)$ , we introduce a *literal edge*  $e_{l,k}$  with latency function  $S_{e_{l,k}}(v_{e_{l,k}}/z_{e_{l,k}}) = v_{e_{l,k}}/z_{e_{l,k}}$  and cost  $\ell_{e_{l,k}} = 1$ . We denote the set of clause edges and literal edges by  $E_K$  and  $E_L$ , respectively.

For each variable  $x_i \in V(\phi)$ , there is a *variable commodity*  $j_{x_i}$  with source  $s_{j_{x_i}}$ , sink  $t_{j_{x_i}}$  and demand  $d_{j_{x_i}} = 1$ . This commodity has two feasible paths, one path uses exclusively the literal edges  $\{e_{x_i,k} : k \in K(\phi)\}$  that correspond to the non-negated variable  $x_i$ , the correspond to the negated variable  $\bar{x}_i$ . In that way, each feasible path of the variable commodity  $j_{x_i}$  corresponds to a **true/false** assignment of the variable  $x_i$ . For each clause  $k = l_k \vee l'_k \vee l''_k$ , we introduce a *clause commodity*  $j_k$  with source  $s_{j_k}$ , sink  $t_{j_k}$  and demand  $d_{j_k} = 1$ . The clause commodity may either choose its corresponding clause edge  $e_k$  or the corresponding literal edges that occur in  $k$ , i.e.,  $e_{l_k,k}$ ,  $e_{l'_k,k}$ , and  $e_{l''_k,k}$ . For notational convenience, we set  $E_k = \{e_{l_k,k}, e_{l'_k,k}, e_{l''_k,k}\}$ . We add some additional edges with latency 0 to obtain a network; see Fig. 1 where these edges are dashed. Note that the problem remains NP-hard, even if we do not allow edges with zero latency, see the full version of this paper.

First, we show that an optimal solution of the so-defined instance of the continuous network design problem  $P$  has total cost less or equal to  $(4 + \epsilon)\kappa + 2\kappa\nu$ , if  $\phi$  has a solution. To this end, let  $\mathbf{y} = (y_{x_i})_{x_i \in V(\phi)}$  be a solution of  $\phi$ . Then, a feasible solution of  $P$  is as follows: For each *positive* literal  $x_i$  that is selected in the solution  $y_i$ , we buy capacity 1 for the corresponding *negative* literal edges  $\{e_{\bar{x}_i,k} : k \in K(\phi)\}$ , and vice versa. Formally, we set

$$z_{a_{l,k}} = \begin{cases} 1, & \text{if } l = x_i \text{ and } y_{x_i} = \text{false}, \\ 1, & \text{if } l = \bar{x}_i \text{ and } y_{x_i} = \text{true}, \\ 0, & \text{otherwise.} \end{cases}$$

For each clause edge  $e_k$ ,  $k \in K(\phi)$ , we buy capacity  $2/\epsilon$ . This particular capacity vector  $\mathbf{z} = (z_e)_{e \in E}$  implies that each variable commodity  $j_{x_i}$  has a unique path of finite length, i.e., the path using the edges corresponding to the *negation* of the corresponding literal in  $\mathbf{y}$ . Using that  $\mathbf{y}$  is a solution of  $\phi$ , we further obtain that for each clause commodity  $j_k$  at least one of the edges in  $E_k$  has capacity zero and, thus, infinite latency. This implies that, in the unique Wardrop equilibrium, the demand of each clause commodity  $j_k$  is routed along the corresponding clause edge  $e_k$ . For the total cost of this solution, we obtain

$$\begin{aligned} C(\mathbf{v}, \mathbf{z}) &= \sum_{e \in E_K} ((4 + v_e/z_e)v_e + (\epsilon/2)^2 z_e) + \sum_{e \in E_L} ((v_e/z_e)v_e + z_e) \\ &= \sum_{e \in E_K} ((4 + \epsilon/2) + (\epsilon/2)) + \frac{1}{2} \sum_{e \in E_L} (1 + 1) = (4 + \epsilon)\kappa + 2\kappa\nu. \end{aligned} \quad (3.1)$$

Hence, an optimal solution has cost not larger than (3.1) if  $\phi$  has a solution.

We proceed to prove that the total cost of an optimal solution are strictly larger than (3.1) if  $\phi$  does *not* admit a solution. Let  $\mathbf{z} = (z_e)_{e \in E}$  be an optimal solution of  $P$  and let  $\mathbf{v} = (v_e)_{e \in E}$  be a corresponding Wardrop flow. We distinguish two cases.

*First case:*  $v_{e_k} > 0$  for all  $k \in K(\phi)$ , i.e., each clause commodity  $j_k$  sends flow over the corresponding clause edge  $e_k$ .

Before we prove the thesis for this case, we need some additional notation. For the Wardrop flow  $v_e$  on edge  $e \in E$ , let  $v_e^V$  and  $v_e^K$  denote the flow on  $e$  that is due to the variable commodities and the clause commodities, respectively. We claim that there is a clause  $\tilde{k} \in K(\phi)$ ,  $\tilde{k} = l_{\tilde{k}} \vee l'_{\tilde{k}} \vee l''_{\tilde{k}}$  such that the flow of the variable commodities on each of the corresponding literal edges in  $E_{\tilde{k}} = \{e_{l_{\tilde{k}}, \tilde{k}}, e_{l'_{\tilde{k}}, \tilde{k}}, e_{l''_{\tilde{k}}, \tilde{k}}\}$  is at least  $1/2$ , i.e.,

$$v_{e_{l_{\tilde{k}}, \tilde{k}}}^V \geq 1/2, \quad v_{e_{l'_{\tilde{k}}, \tilde{k}}}^V \geq 1/2, \quad \text{and} \quad v_{e_{l''_{\tilde{k}}, \tilde{k}}}^V \geq 1/2. \quad (3.2)$$

For a contradiction, let us assume that for each clause  $k = l_k \vee l'_k \vee l''_k$  there is a literal  $l_k^* \in \{l_k, l'_k, l''_k\}$  such that  $v_{e_{l_k^*, k}}^V < 1/2$ . As each variable  $x_i \in V(\phi)$  splits its unit demand between the path consisting of the positive literal edges  $\{e_{x_i, k} : k \in K(\phi)\}$  and the path consisting of the negative literal edges  $\{e_{\bar{x}_i, k} : k \in K(\phi)\}$ , at most one of these two paths is used with a flow strictly smaller than  $1/2$ . Thus, the assignment vector  $\mathbf{y}$  defined as

$$y_{x_i} = \begin{cases} \text{true}, & \text{if } v_e^V < 1/2 \text{ for all } e \in \{e_{x_i, k} : k \in K(\phi)\}, \\ \text{false}, & \text{if } v_e^V < 1/2 \text{ for all } e \in \{e_{\bar{x}_i, k} : k \in K(\phi)\}, \\ \text{true}, & \text{otherwise,} \end{cases}$$

is well-defined. By construction,  $\mathbf{y}$  satisfies all clauses, which is a contradiction to the assumption that no such assignment exists. We conclude that there is a clause  $\tilde{k}$  such that (3.2) holds.

We proceed to bound the total cost of a solution. As  $\mathbf{v}$  is a Wardrop equilibrium in which the clause commodity  $j_{\tilde{k}}$  uses at least partially the clause edge  $e_{\tilde{k}}$ , we further derive that  $\sum_{e \in E_{\tilde{k}}} v_e/z_e \geq v_{e_{\tilde{k}}}/z_{e_{\tilde{k}}} > 4$ . We bound the total cost of the solution  $(\mathbf{v}, \mathbf{z})$  by observing

$$\begin{aligned} C(\mathbf{v}, \mathbf{z}) &= \sum_{e \in E_L} (v_e^2/z_e + z_e) + \sum_{e \in E_K} ((4 + v_e/z_e)v_e + (\epsilon/2)^2 z_e) \\ &\geq \sum_{e \in E_L} \overline{\min}_{z_e \geq 0} (v_e^2/z_e + z_e) + \sum_{e \in E_K} \overline{\min}_{z_e \geq 0} ((4 + v_e/z_e)v_e + (\epsilon/2)^2 z_e), \end{aligned}$$

where we slightly abuse notation by writing  $\overline{\min}_{z_e \geq 0}$  shorthand for  $\min_{z_e \geq 0: \mathbf{v} \in \mathcal{W}(\mathbf{z})}$ . We obtain an upper bound by relaxing  $\overline{\min}_{z_e \geq 0}$  to  $\min_{z_e \geq 0}$  for the edges in  $E_L \setminus E_{\tilde{k}}$  and  $E_K$ . Hence,

$$\begin{aligned} C(\mathbf{v}, \mathbf{z}) &\geq \sum_{e \in E_L \setminus E_{\tilde{k}}} \min_{z_e \geq 0} (v_e^2/z_e + z_e) + \sum_{e \in E_{\tilde{k}}} \overline{\min}_{z_e \geq 0} (v_e^2/z_e + z_e) \\ &\quad + \sum_{e \in E_K} \min_{z_e \geq 0} ((4 + v_e/z_e)v_e + (\epsilon/2)^2 z_e). \end{aligned}$$

Calculating the respective minima, we obtain

$$C(\mathbf{v}, \mathbf{z}) \geq \sum_{e \in E_L \setminus E_{\tilde{k}}} 2v_e + \underbrace{\sum_{e \in E_{\tilde{k}}} \overline{\min}_{z_e \geq 0} (v_e^2/z_e + z_e)}_{\geq 2v_e} + \sum_{e \in E_K} (4 + \epsilon)v_e. \quad (3.3)$$



Each clause commodity  $j_k$  can route its demand either over the clause edge  $e_k$  or over the three literal edges in  $E_k$ . Every fraction of the demand routed over the clause edge contributes  $4 + \epsilon$  to the expression on the right hand side of (3.3) while it contributes at least 6 when routed over the literal edges. Thus, the right hand side of (3.3) is minimized when the clause commodities do not use the literal edges at all. We then obtain

$$\begin{aligned}
C(\mathbf{v}, \mathbf{z}) &\geq \sum_{e \in E_L \setminus E_{\tilde{k}}} 2v_e^V + \sum_{e \in E_{\tilde{k}}} \min_{z_e \geq 0} ((v_e^V)^2 / z_e + z_e) + (4 + \epsilon)|E_K| \\
&= 2\left(\kappa\nu - \sum_{e \in E_{\tilde{k}}} v_e^V\right) + (4 + \epsilon)\kappa + \sum_{e \in E_{\tilde{k}}} \min_{z_e \geq 0} ((v_e^V)^2 / z_e + z_e), \\
&= 2\kappa\nu + (4 + \epsilon)\kappa + \sum_{e \in E_{\tilde{k}}} \min_{z_e \geq 0} ((v_e^V)^2 / z_e + z_e - 2v_e^V), \\
&> 2\kappa\nu + (4 + \epsilon)\kappa + Q,
\end{aligned}$$

where  $Q$  is the solution to the constrained minimization problem

$$\begin{aligned}
Q &= \min_{\substack{v_e^V, z_e > 0 \\ e \in E_{\tilde{k}}}} \sum_{e \in E_{\tilde{k}}} ((v_e^V)^2 / z_e + z_e - 2v_e^V) \\
\text{s.t.: } &\sum_{e \in E_{\tilde{k}}} v_e^V / z_e \geq 4
\end{aligned} \tag{3.4}$$

$$v_e^V \geq 1/2 \text{ for all } e \in E_{\tilde{k}}. \tag{3.5}$$

Side constraint (3.4) is a relaxation of the requirement that  $\mathbf{v}$  is a Wardrop equilibrium as the latency of the literal edges is strictly larger than 4. Side constraint (3.5) is due to the fact that for clause  $\tilde{k}$  the three corresponding literal edges  $e_{l_{\tilde{k}}, \tilde{k}}$ ,  $e_{l'_{\tilde{k}}, \tilde{k}}$ , and  $e_{l''_{\tilde{k}}, \tilde{k}}$  are used with a flow of at least  $1/2$  by the variable commodities. The optimal solution to the constraint optimization problem  $Q$  is equal to  $Q = 1/8$  and is attained for  $v_e^V = 1/2$  and  $z_e = 3/8$  for all  $e \in E_{\tilde{k}}$ . This implies that the total cost of a solution is not smaller than  $(4 + \epsilon)\kappa + 2\kappa\nu + 1/8$ , which finishes the first case of this proof.

*Second case:* There is a clause commodity  $j_{\tilde{k}}$  that does not use its clause edge  $e_{\tilde{k}}$ , i.e.,  $v_{e_{\tilde{k}}} = 0$ . As for first case, we observe

$$C(\mathbf{v}, \mathbf{z}) = \sum_{e \in E_L} (v_e^2 / z_e + z_e) + \sum_{e \in E_K} (4v_e + v_e^2 / z_e + (\epsilon/2)^2 z_e) \geq \sum_{e \in E_L} 2v_e + \sum_{e \in E_K} (4 + \epsilon)v_e.$$

Using that  $j_{\tilde{k}}$  does not use its clause edge, we derive that the flow on the literal edges amounts to  $\nu\kappa + 3$  and we obtain

$$C(\mathbf{v}, \mathbf{z}) \geq 2(\kappa\nu + 3) + (4 + \epsilon)(\kappa - 1) = 2\kappa\nu + (4 + \epsilon)\kappa + 2,$$

which concludes the proof.  $\blacktriangleleft$

With a more involved construction and a more detailed analysis, we can show that CNDP is in fact APX-hard. For this proof, we use a similar construction as in the proof of Theorem 2 but reduce from a specific variant of MAX-3-SAT, which is NP-hard to approximate. Due to space constraints we defer the details to the full version of this paper.

► **Theorem 3.** *The continuous network design problem (CNDP) on directed networks is APX-hard, even if all latency function are affine.*

With a similar construction, we can also show APX-hardness for CNDP on undirected networks as well, see the full version of this paper. For our hardness results, we use instances with different sinks. In contrast, CNDP can be solved efficiently for networks with a single sink.

► **Proposition 3.1.** In networks with only one sink vertex  $t$ , the continuous network design problem (CNDP) can be solved in polynomial time.

## 4 Approximation

Given the APX-hardness of the problem, we study the approximation of CNDP. We first provide a detailed analysis of the approximation guarantees of two different approximation algorithms. Then, as the arguably most interesting result of this section, we provide an improved approximation guarantee for taking the better of the two algorithms. The approximation guarantees proven in this section depend on the set  $\mathcal{S}$  of allowable cost functions and are in fact closely related to the *anarchy value* value  $\alpha(\mathcal{S})$  introduced by Roughgarden [21] and Correa et al. [6]. Intuitively, the anarchy value of a set of latency functions  $\mathcal{S}$  is the worst case ratio between the routing cost of a Wardrop equilibrium and that of a system optimum of an instance in which all latency functions are contained in  $\mathcal{S}$ . Roughgarden [21] and Correa et al. [6] show that  $\alpha(\mathcal{S}) = 1/(1 - \mu(\mathcal{S}))$ , where

$$\mu(\mathcal{S}) = \sup_{S \in \mathcal{S}} \sup_{x \geq 0} \max_{\gamma \in [0,1]} \gamma \cdot \left(1 - \frac{S(\gamma x)}{S(x)}\right). \quad (4.1)$$

For a set  $\mathcal{S}$  of latency functions, we denote by  $\gamma(\mathcal{S})$  the argmaximum  $\gamma$  in (4.1) for which  $\mu(\mathcal{S})$  is achieved. The following lemma gives an alternative representation of  $\mu(\mathcal{S})$ .

► **Lemma 4.** For a latency function  $S$ ,

$$\sup_{x \geq 0} \max_{\gamma \in [0,1]} \left\{ \gamma \left(1 - \frac{S(\gamma x)}{S(x)}\right) \right\} = \sup_{x \geq 0} \left\{ \gamma \cdot \frac{S'(x)x}{S(x) + S'(x)x} : S(x) + S'(x)x = S(x/\gamma) \right\}.$$

**Proof.** The expression  $\sup_{x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{S(\gamma x)}{S(x)}\right)$  is non-negative and strictly positive for  $\gamma \in (0, 1)$ , thus, the inner maximum is attained for  $\gamma \in (0, 1)$ . Hence,  $\gamma$  satisfies the first order optimality conditions

$$\begin{aligned} 0 &= \left(1 - \frac{S(\gamma x)}{S(x)}\right) - \gamma x \cdot \frac{S'(\gamma x)}{S(x)} \\ \Leftrightarrow \quad S(x) &= S(\gamma x) + \gamma x S'(\gamma x) \end{aligned}$$

By substituting  $y = \gamma x$ , we obtain

$$\begin{aligned} &\sup_{x \geq 0} \max_{\gamma \in [0,1]} \gamma \left(1 - \frac{S(\gamma x)}{S(x)}\right) \\ &= \sup_{y \geq 0} \left\{ \gamma \left(1 - \frac{S(y)}{S(y/\gamma)}\right) : \gamma \in [0, 1] \text{ with } S(y/\gamma) = S(y) + S'(y)y \right\} \\ &= \sup_{y \geq 0} \left\{ \gamma \cdot \frac{S'(y)y}{S(y) + S'(y)y} : \gamma \in [0, 1] \text{ with } S(y/\gamma) = S(y) + S'(y)y \right\}, \end{aligned}$$

which proves the lemma. ◀

**Algorithm 1** BRINGToEQUILIBRIUM

---

```

1:  $(\mathbf{v}^*, \mathbf{z}^*) \leftarrow$  solution to (CNDP').
2: for all  $e \in E$  do
3:    $\delta_e \leftarrow v_e^*/z_e^*$ 
4:    $\gamma_e \leftarrow$  solution to  $S_e(\delta_e) + S'_e(\delta_e)\delta_e = S_e(\frac{\delta_e}{\gamma_e})$ 
5:    $z_e \leftarrow \gamma_e z_e^*$ 
6: end for
7: return  $(\mathbf{v}^*, \mathbf{z})$ 

```

---

**Algorithm 2** SCALEUNIFORMLY

---

```

1:  $(\mathbf{v}^*, \mathbf{z}^*) \leftarrow$  solution to (CNDP').
2:  $p \leftarrow C^R(\mathbf{v}^*, \mathbf{z}^*)/C(\mathbf{v}^*, \mathbf{z}^*)$ 
3:  $\lambda \leftarrow \mu(\mathcal{S}) + \sqrt{\mu(\mathcal{S}) \frac{p}{1-p}}$ 
4: Compute Wardrop equilibrium  $\mathbf{v}$ 
   with respect to scaled capacities  $\lambda \mathbf{z}^*$ .
5: return  $(\mathbf{v}, \lambda \mathbf{z}^*)$ 

```

---

**4.1 Two Approximation Algorithms**

The first algorithm that we call BRINGToEQUILIBRIUM (cf. Algorithm 1) was already proposed by Marcotte [18, Section 4.3] and analyzed for monomial latency functions. Our contribution is a more general analysis of BRINGToEQUILIBRIUM that works for arbitrary sets of latency functions  $\mathcal{S}$ , requiring only Assumption 2.1. The second algorithm, that we call SCALEUNIFORMLY (cf. Algorithm 2), is a new algorithm that we introduce in this paper.

For both approximation algorithms, we first compute an optimum solution  $(\mathbf{v}^*, \mathbf{z}^*)$  to a relaxation of CNDP without the equilibrium constraints, i.e., we compute a solution  $(\mathbf{v}^*, \mathbf{z}^*)$  to the problem  $\min_{\mathbf{z} \geq 0} \min_{\mathbf{v} \in \mathcal{F}} \sum_{e \in E} (S_e(v_e/z_e) v_e + z_e \ell_e)$ , which can be done in polynomial time (Proposition 2.2). Then, in both algorithms, we reduce the capacity vector  $\mathbf{z}^*$ , and determine a Wardrop equilibrium for the new capacity vector. The algorithms differ in the way we adjust the capacity vector  $\mathbf{z}^*$ . While in BRINGToEQUILIBRIUM, we reduce the edge capacities individually such that the optimum solution to the relaxation (CNDP') is a Wardrop equilibrium, in SCALEUNIFORMLY, we scale all capacities uniformly by a factor  $\lambda$  (cf. line 2-3) and compute a Wardrop equilibrium for the scaled capacities.

We first show that the approximation guarantee of BRINGToEQUILIBRIUM is at most  $(1 + \mu(\mathcal{S}))$ . For the proof of this result, we use the first order optimality conditions for the vector of capacities  $\mathbf{v}^*$  obtained as a solution to the relaxed problem (CNDP') in combination with the variational inequalities technique used in the price of anarchy literature (e.g. Roughgarden [21] and Correa et al. [6]).

► **Theorem 5.** *The approximation guarantee of BRINGToEQUILIBRIUM is at most  $1 + \mu(\mathcal{S})$ .*

**Proof.** Let  $(\mathbf{v}^*, \mathbf{z}^*)$  be the relaxed solution computed in the first step of BRINGToEQUILIBRIUM. By the necessary Karush-Kuhn-Tucker optimality conditions,  $(\mathbf{v}^*, \mathbf{z}^*)$  satisfies

$$\ell_e = S'_e(v_e^*/z_e^*)(v_e^*/z_e^*)^2, \text{ for all } e \in E \text{ with } z_e^* > 0. \quad (4.2)$$

Eliminating  $\ell_e$  in the statement of the relaxed problem (CNDP') we obtain the following expression for the total cost of the relaxation:

$$C(\mathbf{v}^*, \mathbf{z}^*) = \sum_{e \in E} \left( S_e(v_e^*/z_e^*) + S'_e(v_e^*/z_e^*)(v_e^*/z_e^*) \right) v_e^*. \quad (4.3)$$

For each  $e \in E$  let  $\delta_e = v_e^*/z_e^*$ , if  $z_e^* > 0$ , and  $\delta_e = 0$ , otherwise. We define a new vector of capacities  $\mathbf{z}$  by  $z_e = \gamma_e \cdot z_e^*$ ,  $e \in E$ , where  $\gamma_e \in [0, 1]$  is a solution to the equation

$$S_e(\delta_e) + S'_e(\delta_e) \delta_e = S_e(\delta_e/\gamma_e). \quad (4.4)$$

By Proposition 2.1, the flow  $\mathbf{v}^*$  is a Wardrop flow with respect to  $\mathbf{z}$ . We are interested in

bounding  $C(\mathbf{v}^*, \mathbf{z})$ . To this end, we calculate

$$\begin{aligned} C(\mathbf{v}^*, \mathbf{z}) &= \sum_{e \in E} (S_e(\delta_e/\gamma_e) v_e^* + \ell_e z_e) \stackrel{(4.4)}{=} \sum_{e \in E} \left( (S_e(\delta_e) + S'_e(\delta_e) \delta_e) v_e^* + \gamma_e \ell_e z_e^* \right) \\ &\stackrel{(4.2)}{=} \sum_{e \in E} \left( (S_e(\delta_e) + S'_e(\delta_e) \delta_e) v_e^* + \gamma_e S'_e(\delta_e) \delta_e v_e^* \right). \end{aligned} \quad (4.5)$$

By (4.1), (4.4), and Lemma 4, we have  $\gamma_e S'_e(\delta_e) \delta_e \leq \mu(\mathcal{S}) (S_e(\delta_e) + S'_e(\delta_e) \delta_e)$ . Combining this inequality with (4.5), gives

$$C(\mathbf{v}^*, \mathbf{z}) \leq (1 + \mu(\mathcal{S})) \sum_{e \in E} \left( (S_e(\delta_e) + S'_e(\delta_e) \delta_e) v_e^* \right) \stackrel{(4.3)}{=} (1 + \mu(\mathcal{S})) C(\mathbf{v}^*, \mathbf{z}^*),$$

which completes the proof of the theorem.  $\blacktriangleleft$

We proceed by showing that SCALEUNIFORMLY achieves the same approximation guarantee of  $1 + \mu(\mathcal{S})$ . Recall that SCALEUNIFORMLY first computes a relaxed solution  $(\mathbf{v}^*, \mathbf{z}^*)$ . Then, this relaxed solution is used to compute an optimal scaling factor  $\lambda \leq 1$  with which all capacities are scaled subsequently. The algorithm then returns the scaled capacity vector  $\lambda \mathbf{z}^*$  together with a corresponding Wardrop equilibrium  $\mathbf{v} \in \mathcal{W}(\lambda \mathbf{z}^*)$ .

An (worse) approximation guarantee of 2 can be inferred directly from a bicriteria result of Roughgarden and Tardos [23] who showed that for any instance the routing cost of a Wardrop equilibrium is not worse than a system optimum that ships twice as much flow. This implies that for  $\lambda = 1/2$  we have  $C(\mathbf{v}, \lambda \mathbf{z}^*) \leq 2C(\mathbf{v}^*, \mathbf{z}^*)$ , as claimed.

For the proof of the following result, we take a different road that allows us to express the approximation guarantee of SCALEUNIFORMLY as a function of the parameter  $p$  defined as the fraction of the total cost  $C(\mathbf{v}^*, \mathbf{z}^*)$  of the relaxed solution allotted to the routing costs  $C^R(\mathbf{v}^*, \mathbf{z}^*)$ . This is an important ingredient for the analysis of the best-of-two algorithm.

► **Theorem 6.** *The approximation guarantee of SCALEUNIFORMLY is at most  $(1 + \mu(\mathcal{S}))$ .*

**Proof.** The algorithm first computes an optimum solution  $(\mathbf{v}^*, \mathbf{z}^*)$  of the relaxed problem (CNDP'). Then  $p \in [0, 1]$  is defined as the fraction of  $C(\mathbf{v}^*, \mathbf{z}^*)$  that corresponds to the routing cost  $C^R(\mathbf{v}^*, \mathbf{z}^*)$ , i.e.,  $C^R(\mathbf{v}^*, \mathbf{z}^*) = \sum_{e \in E} S_e(v_e^*/z_e^*) v_e^* = p C(\mathbf{v}^*, \mathbf{z}^*)$ . Now, we define  $\lambda = \mu(\mathcal{S}) + \sqrt{\mu(\mathcal{S}) \frac{p}{1-p}}$  and consider the capacity vector  $\lambda \mathbf{z}^*$ , in which the capacities of the optimal solution to the relaxation are scaled uniformly by  $\lambda$ . Finally, we compute a Wardrop equilibrium with respect to capacities  $\lambda \mathbf{z}^*$ . Let  $\mathbf{v}$  the corresponding equilibrium flow. We now bound the routing and installation cost of  $(\mathbf{v}, \lambda \mathbf{z}^*)$  separately. For the installation cost, we obtain

$$C^Z(\mathbf{v}, \lambda \mathbf{z}^*) = \sum_{e \in E} \lambda \ell_e z_e = \lambda(1 - p) C(\mathbf{v}^*, \mathbf{z})$$

and for the routing cost

$$\begin{aligned} C^R(\mathbf{v}, \lambda \mathbf{z}^*) &= \sum_{e \in E} S_e\left(\frac{v_e}{\lambda z_e^*}\right) v_e \leq \sum_{e \in E} S_e\left(\frac{v_e}{\lambda z_e^*}\right) v_e^* \\ &= p C(\mathbf{v}^*, \mathbf{z}^*) + \sum_{e \in E} \left( S_e\left(\frac{v_e}{\lambda z_e^*}\right) v_e^* - S_e\left(\frac{v_e^*}{z_e^*}\right) v_e^* \right), \end{aligned} \quad (4.6)$$

where the first inequality uses the variational inequality (2.1). We proceed to bound  $S_e(\frac{v_e}{\lambda z_e^*}) v_e^* - S_e(\frac{v_e^*}{z_e^*}) v_e^*$  in terms of the routing cost  $S_e(\frac{v_e}{\lambda z_e^*}) v_e$  for that edge  $e$ . To this end,

note that for each edge  $e \in E$  we have

$$\frac{S_e(\frac{v_e}{\lambda z_e^*})v_e^* - S_e(\frac{v_e}{z_e^*})v_e^*}{S_e(\frac{v_e}{\lambda z_e^*})v_e} \leq \sup_{S \in \mathcal{S}} \sup_{x, y, z \geq 0} \frac{S(\frac{y}{\lambda z})x - S(\frac{x}{z})x}{S(\frac{y}{\lambda z})y} \quad (4.7)$$

$$= \sup_{S \in \mathcal{S}} \sup_{x, y \geq 0} \frac{S(\frac{y}{\lambda})x - S(x)x}{S(\frac{y}{\lambda})y} = \sup_{S \in \mathcal{S}} \sup_{x, y \geq 0} \frac{S(y)x - S(x)x}{S(y)\lambda y}. \quad (4.8)$$

This implies  $y \geq x$  and we may substitute  $x = \gamma y$  with  $\gamma \in [0, 1]$ . We then obtain for each edge  $e \in E$  that

$$\frac{S_e(\frac{v_e}{\lambda z_e^*})v_e^* - S_e(\frac{v_e}{z_e^*})v_e^*}{S_e(\frac{v_e}{\lambda z_e^*})v_e} \leq \sup_{S \in \mathcal{S}} \sup_{y \geq 0} \max_{\gamma \in [0, 1]} \frac{\gamma S(y) - \gamma S(\gamma y)}{\lambda S(y)} \quad (4.9)$$

$$= \sup_{S \in \mathcal{S}} \sup_{y \geq 0} \max_{\gamma \in [0, 1]} \frac{\gamma}{\lambda} \left(1 - \frac{S(\gamma y)}{S(y)}\right) = \frac{\mu(\mathcal{S})}{\lambda}. \quad (4.10)$$

Combining (4.10) and (4.6), we obtain  $C^R(\mathbf{v}, \lambda \mathbf{z}^*) \leq p C(\mathbf{v}^*, \mathbf{z}^*) + \frac{\mu(\mathcal{S})}{\lambda} C^R(\mathbf{v}, \lambda \mathbf{z}^*)$  or, equivalently,  $C^R(\mathbf{v}, \lambda \mathbf{z}^*) \leq \frac{p}{1 - \mu(\mathcal{S})/\lambda} C(\mathbf{v}^*, \mathbf{z}^*)$ . Thus, we can bound the total cost of the outcome of SCALEUNIFORMLY by

$$\begin{aligned} C(\mathbf{v}, \lambda \mathbf{z}^*) &= C^R(\mathbf{v}, \lambda \mathbf{z}^*) + C^Z(\mathbf{v}, \lambda \mathbf{z}^*) \leq \frac{p}{1 - \mu(\mathcal{S})/\lambda} C(\mathbf{v}^*, \mathbf{z}^*) + \lambda(1 - p) C(\mathbf{v}^*, \mathbf{z}^*) \\ &= \lambda \left( \frac{p}{\lambda - \mu(\mathcal{S})} + 1 - p \right) C(\mathbf{v}^*, \mathbf{z}^*). \end{aligned}$$

Since  $\lambda = \mu(\mathcal{S}) + \sqrt{\mu(\mathcal{S}) \frac{p}{1-p}}$ , we obtain

$$\frac{C(\mathbf{v}, \lambda \mathbf{z}^*)}{C(\mathbf{v}^*, \mathbf{z}^*)} \leq p + 2\sqrt{p(1-p)\mu(\mathcal{S})} + \mu(\mathcal{S})(1-p) = (\sqrt{p} + \sqrt{\mu(\mathcal{S})(1-p)})^2. \quad (4.11)$$

Elementary calculus shows that  $(\sqrt{p} + \sqrt{\mu(\mathcal{S})(1-p)})^2$  attains its maximum at  $p = \frac{1}{1+\mu(\mathcal{S})}$ . Substituting this value into (4.11) gives  $C(\mathbf{v}, \lambda \mathbf{z}^*)/C(\mathbf{v}^*, \mathbf{z}^*) \leq 1 + \mu(\mathcal{S})$ , as claimed.  $\blacktriangleleft$

For particular sets  $\mathcal{S}$  of latency functions, we compute upper bounds on  $\mu(\mathcal{S})$  in order to obtain an explicit upper bound on the approximation guarantees of BRINGTOEQUILIBRIUM and SCALEUNIFORMLY. We then obtain the following corollary of Theorem 5 and Theorem 6.

► **Corollary 7.** *For a set  $\mathcal{S}$  of latency functions satisfying Assumption 2.1, the approximation guarantee of BRINGTOEQUILIBRIUM and SCALEUNIFORMLY is at most*

- (a) 2, without further requirements on  $\mathcal{S}$ .
- (b) 5/4, if  $\mathcal{S}$  contains concave latencies only,
- (c)  $1 + \frac{\Delta}{\Delta+1} \left(\frac{1}{\Delta+1}\right)^{1/\Delta}$ , if  $\mathcal{S}$  contains only polynomials with non-negative coefficients and degree at most  $\Delta$ , i.e., each  $S \in \mathcal{S}$  is of the form  $S(x) = \sum_{j=0}^{\Delta} a_j x^j$  with  $a_j \geq 0$  for all  $j$ .

## 4.2 Best-of-Two Approximation

In this section we show that although both BRINGTOEQUILIBRIUM and SCALEUNIFORMLY achieve an approximation guarantee of  $(1 + \mu(\mathcal{S}))$  taking the better of the two algorithms we obtain a strictly better performance guarantee.

The key idea of the proof is to extend the analysis of the BRINGTOEQUILIBRIUM algorithm in order to express its approximation guarantee as a function of the parameter  $p$  that measures the proportion of the routing cost in the total cost of a relaxed solution. This allows us to determine the worst-case  $p$  for which the approximation guarantee of the both algorithm is maximized.

► **Theorem 8.** *Taking the better solution of BRINGTOEQUILIBRIUM and SCALEUNIFORMLY has an approximation guarantee of at most  $\frac{(\gamma(\mathcal{S})+\mu(\mathcal{S})+1)^2}{(\gamma(\mathcal{S})+\mu(\mathcal{S})+1)^2-4\mu(\mathcal{S})\gamma(\mathcal{S})}$ , which is strictly smaller than  $1 + \mu(\mathcal{S})$ .*

**Proof.** Recall from (4.11) that the approximation guarantee of the algorithm SCALEUNIFORMLY is  $(\sqrt{p} + \sqrt{\mu(\mathcal{S})(1-p)})^2$ , where  $p = C^R(\mathbf{v}^*, \mathbf{z}^*)/C(\mathbf{v}^*, \mathbf{z}^*)$ . We extend our analysis of BRINGTOEQUILIBRIUM using this parameter  $p$ . With the notation in Theorem 5, by (4.5), BRINGTOEQUILIBRIUM returns a feasible solution  $(\mathbf{v}^*, \mathbf{z})$  with

$$\begin{aligned} C(\mathbf{v}^*, \mathbf{z}) &= \sum_{e \in E} \left( (S_e(\delta_e) + S'_e(\delta_e) \delta_e) v_e^* + \gamma_e S'_e(\delta_e) \delta_e v_e^* \right) \\ &= p C(\mathbf{v}^*, \mathbf{z}^*) + \sum_{e \in E} S'_e(\delta_e) \delta_e v_e^* (1 + \gamma_e) \\ &\leq p C(\mathbf{v}^*, \mathbf{z}^*) + (1 + \gamma(\mathcal{S})) \sum_{e \in E} S'_e(\delta_e) \delta_e v_e^* \\ &= p C(\mathbf{v}^*, \mathbf{z}^*) + (1 + \gamma(\mathcal{S}))(1 - p) C(\mathbf{v}^*, \mathbf{z}^*) \\ &= (1 + \gamma(\mathcal{S})(1 - p)) C(\mathbf{v}^*, \mathbf{z}^*). \end{aligned}$$

Thus, by taking the best of the two heuristics, we obtain an approximation guarantee of

$$\max_{p \in (0,1)} \min \left\{ 1 + \gamma(\mathcal{S})(1 - p), \left( \sqrt{p} + \sqrt{\mu(\mathcal{S})(1 - p)} \right)^2 \right\}.$$

The maximum of this expression is attained for

$$p = p^* := \frac{(\gamma(\mathcal{S}) - \mu(\mathcal{S}) + 1)^2}{(\gamma(\mathcal{S}) - \mu(\mathcal{S}) + 1)^2 + 4\mu(\mathcal{S})} \quad (4.12)$$

which yields the claimed improved upper bound. ◀

It is not necessary to run both approximation algorithms to get this approximation guarantee. After computing the optimum solution to the relaxation (CNDP'), we can determine the value for  $p = C^R(\mathbf{v}^*, \mathbf{z}^*)/C(\mathbf{v}^*, \mathbf{z}^*)$  and proceed with SCALEUNIFORMLY if  $p \leq p^*$  (cf. (4.12)) and with BRINGTOEQUILIBRIUM otherwise.

For particular sets  $\mathcal{S}$  of latency functions, we evaluate  $\mu(\mathcal{S})$  and  $\gamma(\mathcal{S})$  and obtain the following corollary of Theorem 8.

► **Corollary 9.** *For a set  $\mathcal{S}$  of latency functions satisfying Assumption 2.1, the approximation guarantee in Theorem 8 is at most*

- (a)  $9/5$ , without further requirements on  $\mathcal{S}$ ,
- (b)  $49/41 \approx 1.195$ , if  $\mathcal{S}$  contains concave latencies only.
- (c)  $1 + \frac{4\Delta(\Delta+1)}{2(2\Delta+1)(\Delta+1)^{1+1/\Delta} + (\Delta+1)^{2(1+1/\Delta)} + 1}$ , if  $\mathcal{S}$  contains only polynomials with non-negative coefficients and degree at most  $\Delta$ , i.e., every  $S \in \mathcal{S}$  is of the form  $S(x) = \sum_{j=0}^{\Delta} a_j x^j$  with  $a_j \geq 0$  for all  $j$ .

## 5 Conclusion

We reconsidered the classical continuous network design problem (CNDP) and established the first hardness result for CNDP. Further, we provided a general approximation guarantee for an algorithm studied by Marcotte [18] depending on the set of allowed cost functions which is related to the *anarchy value* of the set of cost functions. We then showed that the

approximation of the problem can be improved by taking the best of that algorithm and another approximation algorithm that we propose.

In the transportation literature, further variants of CNDP have been investigated. One such example are situations in which the network designer is only interested in minimizing total travel time but investments are restricted, e.g., by budget constraints. More generally, suppose there is a convex function  $g : \mathbb{R}^m \rightarrow \mathbb{R}^k$ ,  $k \in \mathbb{N}$  such that for any feasible solution  $\mathbf{z}$  the condition  $g(\mathbf{z}) \leq \mathbf{0}$  must be satisfied. The function  $g$ , for instance, can represent edge-specific budget constraints  $\ell_e z_e \leq B_e$  for  $e \in E$  and/or a global budget constraint  $\sum_{e \in E} \ell_e z_e \leq B$ . We arrive at the following budgeted continuous network design problem (bCNDP):

$$\min_{\mathbf{z} \geq \mathbf{0}} \min_{\mathbf{v} \in \mathcal{W}(\mathbf{z})} \sum_{e \in E} S_e(v_e/z_e) v_e \quad \text{s.t. : } g(\mathbf{z}) \leq \mathbf{0}. \quad (\text{bCNDP})$$

Using existing results from the literature [6, 21], we can show a  $4/3$ -approximation for affine latencies, and that there is no polynomial  $(4/3 - \epsilon)$ -approximation algorithm with  $\epsilon > 0$  unless  $\text{P} = \text{NP}$ , see the full version of this paper. For proving the lower bound, we use edge-specific budget constraints and mimic a construction from Roughgarden [22]. It is an interesting open problem whether such a lower bound can also be achieved if we allow only a global budget constraint.

---

## References

- 1 M. Abdulaal and L. J. LeBlanc. Continuous equilibrium network design models. *Transportation Res. Part B*, 13(B):19–32, 1979.
- 2 R. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice-Hall, Upper Saddle River, NJ, USA, 1993.
- 3 M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics and Transportation*. Yale University Press, New Haven, CT, USA, 1956.
- 4 U. Bhaskar, K. Ligett, and L. J. Schulman. Network improvement for equilibrium routing. In *Proc. 17th Int. Conf. on Integer Programming and Combinatorial Optimization (IPCO)*, pages 138–149, 2014.
- 5 B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of Oper. Res.*, 153(1):235–256, 2007.
- 6 J. Correa, A. Schulz, and N. Stier-Moses. Selfish routing in capacitated networks. *Math. Oper. Res.*, 29(4):961–976, 2004.
- 7 S. C. Dafermos. *Traffic assignment and resource allocation in transportation networks*. PhD thesis, John Hopkins University, Baltimore, MD, 1968.
- 8 S. C. Dafermos. Traffic equilibrium and variational inequalities. *Transportation Sci.*, 14:42–54, 1980.
- 9 G. B. Dantzig, R. P. Harvey, Z. F. Lansdowne, D. W. Robinson, and S. F. Maier. Formulating and solving the network design problem by decomposition. *Transportation Res. Part B*, 13(1):5–17, 1979.
- 10 T. L. Friesz. Transportation network equilibrium, design and aggregation: Key developments and research opportunities. *Transportation Res. Part A*, 17(B):411–426, 1985.
- 11 M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer, Berlin, Germany, 2nd edition, 1993.
- 12 Y. A. Korilis, A. A. Lazar, and A. Orda. Architecting noncooperative networks. *IEEE J. Sel. Area Commun.*, 13(7):1241–1251, 1995.

- 13 Y. A. Korilis, A. A. Lazar, and A. Orda. Avoiding the Braess paradox in noncooperative networks. *J. Appl. Probab.*, 36(1):211–222, 1999.
- 14 L. Libman and A. Orda. The designer’s perspective to atomic noncooperative networks. *IEEE/ACM Trans. Networking*, 7(6):875–884, 1999.
- 15 H. Lin, T. Roughgarden, É. Tardos, and A. Walkover. Stronger bounds on Braess’s paradox and the maximum latency of selfish routing. *SIAM J. Comput.*, 25(4):1667–1686, 2011.
- 16 Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical programs with equilibrium constraints*. Cambridge University Press, Cambridge, UK, 1996.
- 17 T. L. Magnanti and R. T. Wong. Network design and transportation planning: Models and algorithms. *Transportation Sci.*, 18(1):1–55, 1984.
- 18 P. Marcotte. Network design problem with congestion effects: A case of bilevel programming. *Math. Program.*, 34:142–162, 1986.
- 19 P. Marcotte and G. Marquis. Efficient implementation of heuristics for the continuous network design problem. *Annals of Oper. Res.*, 34:163–176, 1992.
- 20 L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker. On selfish routing in Internet-like environments. *IEEE/ACM Trans. Networking*, 14(4):725–738, 2006.
- 21 T. Roughgarden. The price of anarchy is independent of the network topology. *J. Comput. System Sci.*, 67:341–364, 2002.
- 22 T. Roughgarden. On the severity of Braess’s paradox: Designing networks for selfish users is hard. *J. Comput. System Sci.*, 72(5):922–953, 2006.
- 23 T. Roughgarden and É. Tardos. How bad is selfish routing? *J. ACM*, 49(2):236–259, 2002.
- 24 M. J. Smith. The existence, uniqueness and stability of traffic equilibria. *Transportation Res.*, 13(B):295–304, 1979.
- 25 U.S. Bureau of Public Roads. *Traffic assignment manual*. U.S. Department of Commerce, Urban Planning Division, Washington, DC, 1964.
- 26 G. Valiant and T. Roughgarden. Braess’s paradox in large random graphs. *Random Structures Algorithms*, 37(4):495–515, 2010.
- 27 H. Yang and M. G. H. Bell. Models and algorithms for road network design: a review and some new developments. *Transport Reviews*, 18(3):257–278, 1998.